

## Task#1:

Q.1.

<pre>Command Prompt  C:\Users\Syed Arham Ahmed&gt;nslookup nu.edu.pk Server: gpon.net Address: 192.168.1.1  Non-authoritative answer: Name: nu.edu.pk Address: 203.124.44.78</pre>	<div>Syed Arham Ahmed BS-CY-B 22i-1552</div>
--	--

Q.2.

<pre>C:\Users\Syed Arham Ahmed&gt;nslookup flex.nu.edu.pk Server: gpon.net Address: 192.168.1.1  Non-authoritative answer: Name: flex.nu.edu.pk Address: 115.186.60.84</pre>	<div>Syed Arham Ahmed BS-CY-B 22i-1552</div>
--	--

Q.3.

<pre>C:\Users\Syed Arham Ahmed&gt;nslookup -type=NS nu.edu.pk Server: gpon.net Address: 192.168.1.1  Non-authoritative answer: nu.edu.pk      nameserver = n1.comsats.net.pk nu.edu.pk      nameserver = n2.comsats.net.pk</pre>	<div>Syed Arham Ahmed BS-CY-B 22i-1552</div>
--	--

Q.4.

<pre>C:\Users\Syed Arham Ahmed&gt;nslookup -type=NS flex.nu.edu.pk Server: gpon.net Address: 192.168.1.1  nu.edu.pk primary name server = n1.comsats.net.pk responsible mail addr = hosting.comsats.net.pk serial = 2024030401 refresh = 10800 (3 hours) retry = 3600 (1 hour) expire = 604800 (7 days) default TTL = 86400 (1 day)</pre>	<div>Syed Arham Ahmed BS-CY-B 22i-1552</div>
---	--

Q.5.

<pre>C:\Users\Syed Arham Ahmed&gt;nslookup n1.comsats.net.pk Server: gpon.net Address: 192.168.1.1  Non-authoritative answer: Name: n1.comsats.net.pk Address: 210.56.11.130</pre>	<div>Syed Arham Ahmed BS-CY-B 22i-1552</div>
--	--

Q.6.

```
C:\Users\Syed Arham Ahmed>nslookup -query=mx nu.edu.pk
Server: gpon.net
Address: 192.168.1.1
```

Non-authoritative answer:

```
nu.edu.pk      MX preference = 10, mail exchanger = aspmx5.googlemail.com
nu.edu.pk      MX preference = 5, mail exchanger = alt2.aspmx.l.google.com
nu.edu.pk      MX preference = 5, mail exchanger = alt1.aspmx.l.google.com
nu.edu.pk      MX preference = 10, mail exchanger = aspmx2.googlemail.com
nu.edu.pk      MX preference = 0, mail exchanger = aspmx.l.google.com
nu.edu.pk      MX preference = 10, mail exchanger = aspmx4.googlemail.com
nu.edu.pk      MX preference = 10, mail exchanger = aspmx3.googlemail.com
```

+

Syed Arham Ahmed  
BS-CY-B  
22i-1552

Q.7.

```
C:\Users\Syed Arham Ahmed>nslookup -type=NS giki.edu.pk
Server: gpon.net
Address: 192.168.1.1
```

Non-authoritative answer:

```
giki.edu.pk    nameserver = ns2.giki.edu.pk
giki.edu.pk    nameserver = ns.giki.edu.pk
giki.edu.pk    nameserver = localns2.giki.edu.pk
giki.edu.pk    nameserver = localns1.giki.edu.pk
giki.edu.pk    nameserver = ns1.giki.edu.pk
```

+

Syed Arham Ahmed  
BS-CY-B  
22i-1552

```
C:\Users\Syed Arham Ahmed>nslookup -type=soa giki.edu.pk
Server: gpon.net
Address: 192.168.1.1
```

Non-authoritative answer:

```
giki.edu.pk
    primary name server = ns.giki.edu.pk
    responsible mail addr = shakir.giki.edu.pk
    serial = 958
    refresh = 900 (15 mins)
    retry = 600 (10 mins)
    expire = 86400 (1 day)
    default TTL = 3600 (1 hour)
```

+

Syed Arham Ahmed  
BS-CY-B  
22i-1552

```
C:\Users\Syed Arham Ahmed>nslookup ns2.giki.edu.pk
Server: gpon.net
Address: 192.168.1.1
```

Non-authoritative answer:

```
Name: ns2.giki.edu.pk
Address: 119.159.235.50
```

+

Syed Arham Ahmed  
BS-CY-B  
22i-1552

Q.8.

```
C:\Users\Syed Arham Ahmed>nslookup nu.edu.pk 1.1.1.1
Server: one.one.one.one
Address: 1.1.1.1
```

Non-authoritative answer:

```
Name: nu.edu.pk
Address: 203.124.44.78
```

+

Syed Arham Ahmed  
BS-CY-B  
22i-1552

```
C:\Users\Syed Arham Ahmed>nslookup nu.edu.pk 8.8.8.8
Server:      dns.google
Address:     8.8.8.8

Non-authoritative answer:
Name:        nu.edu.pk
Address:     203.124.44.78
```

## Task#2:

Q.1.

```
(arham@kali)~$ dig nu.edu.pk

; <<>> DiG 9.18.16-1-Debian <<>> nu.edu.pk
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 13120
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;nu.edu.pk.                IN      A

;; ANSWER SECTION:
nu.edu.pk.                 5991    IN      A      203.124.44.78

;; Query time: 4 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Mar 16 15:01:25 PKT 2024
;; MSG SIZE rcvd: 43
```

Syed Arham Ahmed  
BS-CY-B  
22i-1552

The record type is A, which stands for Address Record.

Q.2.

```
(arham@kali)-[~]
$ dig ns

; <<> DiG 9.18.16-1-Debian <<> ns
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 40559
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1220
; COOKIE: 47d622e7f3673f9533afbb8565f56eb287577c5421cb1371 (good)
;; QUESTION SECTION:
;.                IN      NS

;; ANSWER SECTION:
.                 387143 IN      NS      b.root-servers.net.
.                 387143 IN      NS      h.root-servers.net.
.                 387143 IN      NS      i.root-servers.net.
.                 387143 IN      NS      e.root-servers.net.
.                 387143 IN      NS      d.root-servers.net.
.                 387143 IN      NS      g.root-servers.net.
.                 387143 IN      NS      a.root-servers.net.
.                 387143 IN      NS      f.root-servers.net.
.                 387143 IN      NS      c.root-servers.net.
.                 387143 IN      NS      m.root-servers.net.
.                 387143 IN      NS      j.root-servers.net.
.                 387143 IN      NS      k.root-servers.net.
.                 387143 IN      NS      l.root-servers.net.

;; Query time: 16 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Mar 16 15:04:34 PKT 2024
;; MSG SIZE rcvd: 267
```

We can see 13 root servers after command dig ns.

Syed Arham Ahmed  
BS-CY-B  
22i-1552

Q.3.

```
(arham@kali)-[~]
$ dig +short NS nu.edu.pk

n2.comsats.net.pk. n1.comsats.net.pk.
n1.comsats.net.pk.

(arham@kali)-[~]
$ dig n2.comsats.net.pk nu.edu.pk

; <<>> DiG 9.18.16-1-Debian <<>> n2.comsats.net.pk nu.edu.pk
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 63845
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1220
;; COOKIE: 65b4ccb66bdeea0b1ba33b4865f57d71267007bc2a822416 (good)
;; QUESTION SECTION:
;n2.comsats.net.pk.                IN      A

;; ANSWER SECTION:
n2.comsats.net.pk. 9541     IN      A      210.56.11.131

;; Query time: 48 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Mar 16 16:07:31 PKT 2024
;; MSG SIZE rcvd: 90

;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 39971
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;nu.edu.pk.                IN      A

;; ANSWER SECTION:
nu.edu.pk. 2027     IN      A      203.124.44.78

;; Query time: 4 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Mar 16 16:07:31 PKT 2024
;; MSG SIZE rcvd: 43
```

Syed Arham Ahmed  
BS-CY-B  
22i-1552

## Q.4.

```
(arham@kali)-[~]
$ dig +norecurse NS edu.

; <<>> DiG 9.18.16-1-Debian <<>> +norecurse NS edu.
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 19726
;; flags: qr ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1220
; COOKIE: a3277bfab62f3d3280c1471965f57f558232c257c9246720 (good)
;; QUESTION SECTION:
;edu.      IN      NS

;; ANSWER SECTION:
edu.      37287   IN      NS      i.edu-servers.net.
edu.      37287   IN      NS      f.edu-servers.net.
edu.      37287   IN      NS      m.edu-servers.net.
edu.      37287   IN      NS      j.edu-servers.net.
edu.      37287   IN      NS      g.edu-servers.net.
edu.      37287   IN      NS      k.edu-servers.net.
edu.      37287   IN      NS      a.edu-servers.net.
edu.      37287   IN      NS      c.edu-servers.net.
edu.      37287   IN      NS      b.edu-servers.net.
edu.      37287   IN      NS      d.edu-servers.net.
edu.      37287   IN      NS      e.edu-servers.net.
edu.      37287   IN      NS      l.edu-servers.net.
edu.      37287   IN      NS      h.edu-servers.net.

;; Query time: 8 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Mar 16 16:15:36 PKT 2024
;; MSG SIZE rcvd: 283
```

## Q.5.

```
(arham@kali)-[~]
$ dig +norecurse NS pk. @a.root-servers.net

; <<>> DiG 9.18.16-1-Debian <<>> +norecurse NS pk. @a.root-servers.net
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 19491
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ADDITIONAL: 7

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;pk.      IN      NS

;; AUTHORITY SECTION:
pk.      172800  IN      NS      root-c1.pknict.pk.
pk.      172800  IN      NS      root-s.pknict.pk.
pk.      172800  IN      NS      root-c2.pknict.pk.
pk.      172800  IN      NS      root-e.pknict.pk.

;; ADDITIONAL SECTION:
root-c1.pknict.pk. 172800 IN A      185.159.197.160
root-c1.pknict.pk. 172800 IN AAAA   2620:10a:80aa::160
root-s.pknict.pk.  172800 IN A      119.81.34.90
root-c2.pknict.pk. 172800 IN A      185.159.198.160
root-c2.pknict.pk. 172800 IN AAAA   2620:10a:80ab::160
root-e.pknict.pk.  172800 IN A      107.6.178.178

;; Query time: 272 msec
;; SERVER: 198.41.0.4#53(a.root-servers.net) (UDP)
;; WHEN: Sat Mar 16 16:29:43 PKT 2024
;; MSG SIZE rcvd: 243
```

## Task#3:

### Q.1.

The top screenshot shows a Wireshark packet capture of a web browser request. The packet list shows a GET request for /favicons.ico. The packet details pane shows the Hypertext Transfer Protocol section. The packet bytes pane shows the raw data. The browser screenshot shows the URL http://sneaindia.com/images/tw.png.

No.	Time	Source	Destination	Protocol	Length	Info
37	7.813681	192.168.1.6	103.240.91.101	HTTP	437	GET /favicons.ico HTTP/1.1
50	8.065491	103.240.91.101	192.168.1.6	TCP	54	80 → 58402 [ACK] Seq=1 Ack=384 Win=501 Len=0
51	8.065491	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=1 Ack=384 Win=501 Len=1452
52	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=1453 Ack=384 Win=501 Len=0
53	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=2905 Ack=384 Win=501 Len=14
54	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=4357 Ack=384 Win=501 Len=0
55	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=5809 Ack=384 Win=501 Len=14
56	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=7261 Ack=384 Win=501 Len=0
57	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=8713 Ack=384 Win=501 Len=14
58	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=10165 Ack=384 Win=501 Len=0
59	8.066153	103.240.91.101	192.168.1.6	HTTP	955	HTTP/1.1 200 OK (image/vnd.microsoft.icon)
60	8.066194	192.168.1.6	103.240.91.101	TCP	54	58402 → 80 [ACK] Seq=384 Ack=12518 Win=516 Len=0
105	13.070569	103.240.91.101	192.168.1.6	TCP	54	80 → 58402 [FIN, ACK] Seq=12518 Ack=384 Win=501 Len=0
106	13.070613	192.168.1.6	103.240.91.101	TCP	54	58402 → 80 [ACK] Seq=384 Ack=12519 Win=516 Len=0

The bottom screenshot shows a Wireshark packet capture of a web browser request. The packet list shows a GET request for /favicons.ico. The packet details pane shows the Hypertext Transfer Protocol section. The packet bytes pane shows the raw data. The browser screenshot shows the URL http://sneaindia.com/images/tw.png.

No.	Time	Source	Destination	Protocol	Length	Info
37	7.813681	192.168.1.6	103.240.91.101	HTTP	437	GET /favicons.ico HTTP/1.1
50	8.065491	103.240.91.101	192.168.1.6	TCP	54	80 → 58402 [ACK] Seq=1 Ack=384 Win=501 Len=0
51	8.065491	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=1 Ack=384 Win=501 Len=1452
52	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=1453 Ack=384 Win=501 Len=0
53	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=2905 Ack=384 Win=501 Len=14
54	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=4357 Ack=384 Win=501 Len=0
55	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=5809 Ack=384 Win=501 Len=14
56	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=7261 Ack=384 Win=501 Len=0
57	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [ACK] Seq=8713 Ack=384 Win=501 Len=14
58	8.066153	103.240.91.101	192.168.1.6	TCP	1506	80 → 58402 [PSH, ACK] Seq=10165 Ack=384 Win=501 Len=0
59	8.066153	103.240.91.101	192.168.1.6	HTTP	955	HTTP/1.1 200 OK (image/vnd.microsoft.icon)
60	8.066194	192.168.1.6	103.240.91.101	TCP	54	58402 → 80 [ACK] Seq=384 Ack=12518 Win=516 Len=0
105	13.070569	103.240.91.101	192.168.1.6	TCP	54	80 → 58402 [FIN, ACK] Seq=12518 Ack=384 Win=501 Len=0
106	13.070613	192.168.1.6	103.240.91.101	TCP	54	58402 → 80 [ACK] Seq=384 Ack=12519 Win=516 Len=0
427	35.825197	192.168.1.6	103.240.91.101	TCP	54	58402 → 80 [FIN, ACK] Seq=384 Ack=12519 Win=516 Len=0
437	36.080379	103.240.91.101	192.168.1.6	TCP	54	80 → 58402 [ACK] Seq=12519 Ack=385 Win=501 Len=0

So, the difference that I noticed now is that when I re-pasted the URL is that it did not generate more than 2 more packets, this means that cache was used to fetch the URL contents, and it did



## Syed Arham Ahmed\_22i-1552\_BS-CY-B\_CNET\_Assignment#2

not do TCP handshake again as well saving us time when re opening the URL. I can also see the HTML code of the still webpage if I follow and open HTTP stream of the packets.

The image displays a network traffic capture using Wireshark on the left and a web browser on the right. The Wireshark interface shows a list of captured packets, with the selected packet (No. 1398) being an HTTP GET request for a static image. The packet details pane shows the structure of the HTTP request, including the method (GET), URI, and headers. The packet bytes pane shows the raw data in hexadecimal and ASCII. The browser window on the right shows the sneaindia.com website, which is a page for the Sanchar Nigam Executives Association. A yellow sticky note is placed over the browser window, containing the user's name, ID, and course details.

No.	Time	Source	Destination	Protocol	Length	Info
1356	70.088833	192.168.1.6	103.240.91.101	TCP	54	58426 → 80 [ACK] Seq=1143 Ack=128286 Win=13205
1366	70.339630	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=128286 Ack=1143 Win=6412
1367	70.339630	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=129738 Ack=1143 Win=
1368	70.339630	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=131190 Ack=1143 Win=6412
1369	70.339630	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=132642 Ack=1143 Win=
1370	70.339688	192.168.1.6	103.240.91.101	TCP	54	58426 → 80 [ACK] Seq=1143 Ack=134094 Win=13205
1371	70.340109	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=134094 Ack=1143 Win=6412
1372	70.340109	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=135546 Ack=1143 Win=
1373	70.340109	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=136998 Ack=1143 Win=6412
1374	70.340109	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=138450 Ack=1143 Win=
1375	70.340136	192.168.1.6	103.240.91.101	TCP	54	58426 → 80 [ACK] Seq=1143 Ack=139902 Win=13205
1376	70.340825	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=139902 Ack=1143 Win=6412
1377	70.340825	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=141354 Ack=1143 Win=
1378	70.340825	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=142806 Ack=1143 Win=6412
1379	70.340825	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=144258 Ack=1143 Win=
1380	70.340859	192.168.1.6	103.240.91.101	TCP	54	58426 → 80 [ACK] Seq=1143 Ack=145710 Win=13205
1381	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=145710 Ack=1143 Win=6412
1382	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=147162 Ack=1143 Win=
1383	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=148614 Ack=1143 Win=6412
1384	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=150066 Ack=1143 Win=
1385	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=151518 Ack=1143 Win=6412
1386	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=152970 Ack=1143 Win=
1387	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=154422 Ack=1143 Win=
1388	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=155874 Ack=1143 Win=6412
1389	70.341776	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=157326 Ack=1143 Win=
1390	70.341833	192.168.1.6	103.240.91.101	TCP	54	58426 → 80 [ACK] Seq=1143 Ack=158778 Win=13205
1391	70.342157	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [ACK] Seq=158778 Ack=1143 Win=6412
1392	70.342157	103.240.91.101	192.168.1.6	TCP	54	80 → 58426 [PSH, ACK] Seq=160230 Ack=1143 Win=
1393	70.342186	192.168.1.6	103.240.91.101	TCP	54	58426 → 80 [ACK] Seq=1143 Ack=161682 Win=13205
1397	70.591666	103.240.91.101	192.168.1.6	HTTP	461	HTTP/1.1 200 OK (JPEG JFIF image)
1398	70.638610	192.168.1.6	103.240.91.101	TCP	54	58426 → 80 [ACK] Seq=1143 Ack=162089 Win=13151

Frame 37: 437 bytes on wire (3496 bits), 437 bytes captured (3496 bits) on interface 0  
Ethernet II, Src: IntelCor\_8e:96:7b (a0:e0:6f:6e:2e:69), Dst: 192.168.1.6 (08:00:27:00:00:00)  
Internet Protocol Version 4, Src: 192.168.1.6, Dst: 103.240.91.101  
Transmission Control Protocol, Src Port: 58426, Dst Port: 80  
Hypertext Transfer Protocol

0030 02 04 86 9d 00 00 47 45 54 20 2f 66 61 76 69 63 .....GE T /favic  
0040 6f 6e 2e 69 63 6f 20 48 54 50 2f 31 2e 31 0d on.ico H TTP/1.1  
0050 0a 4b 6f 73 74 3a 20 73 6e 65 61 69 6e 64 69 61 .Host: sneaindia  
0060 2e 63 6f 6d 0a 43 6f 6e 65 63 74 69 6f 6e .com: Connection  
0070 3a 20 6b 65 65 70 2d 61 6c 69 76 65 0d 0a 55 73 : keep-a live-Us  
0080 65 72 2d 41 67 65 74 3a 20 4d 6f 7a 69 6c 6c er-Agent : Mozilla

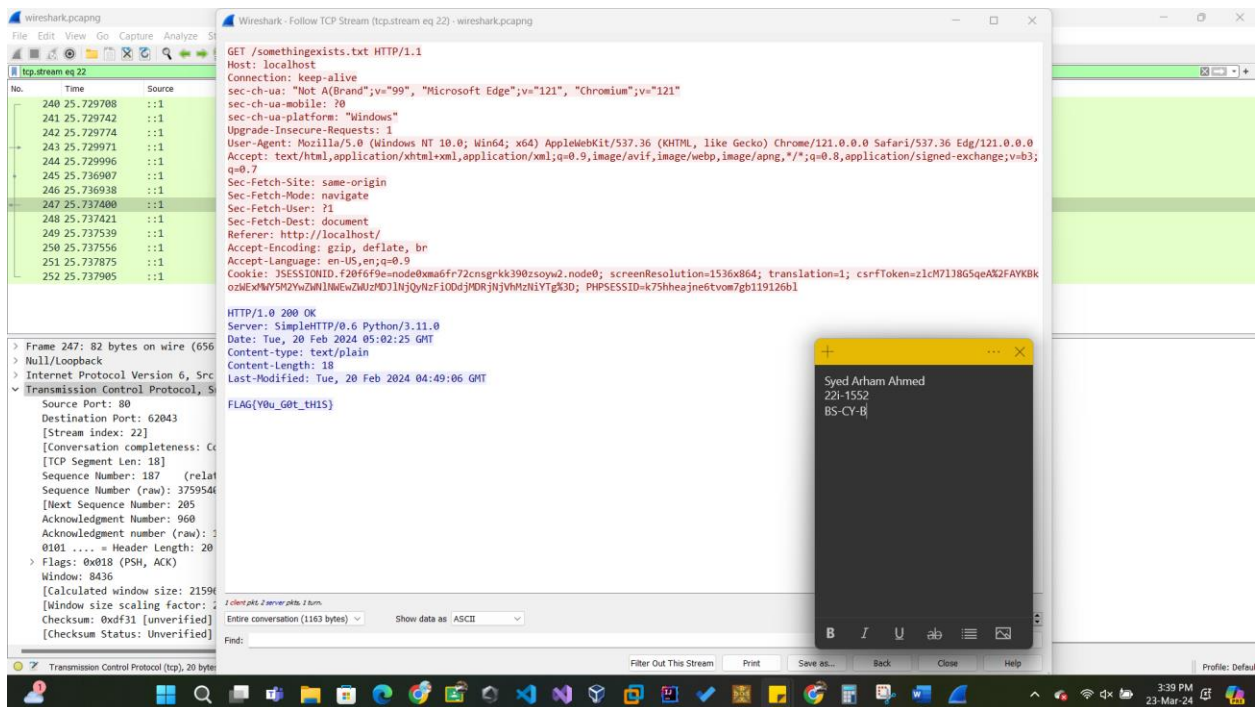
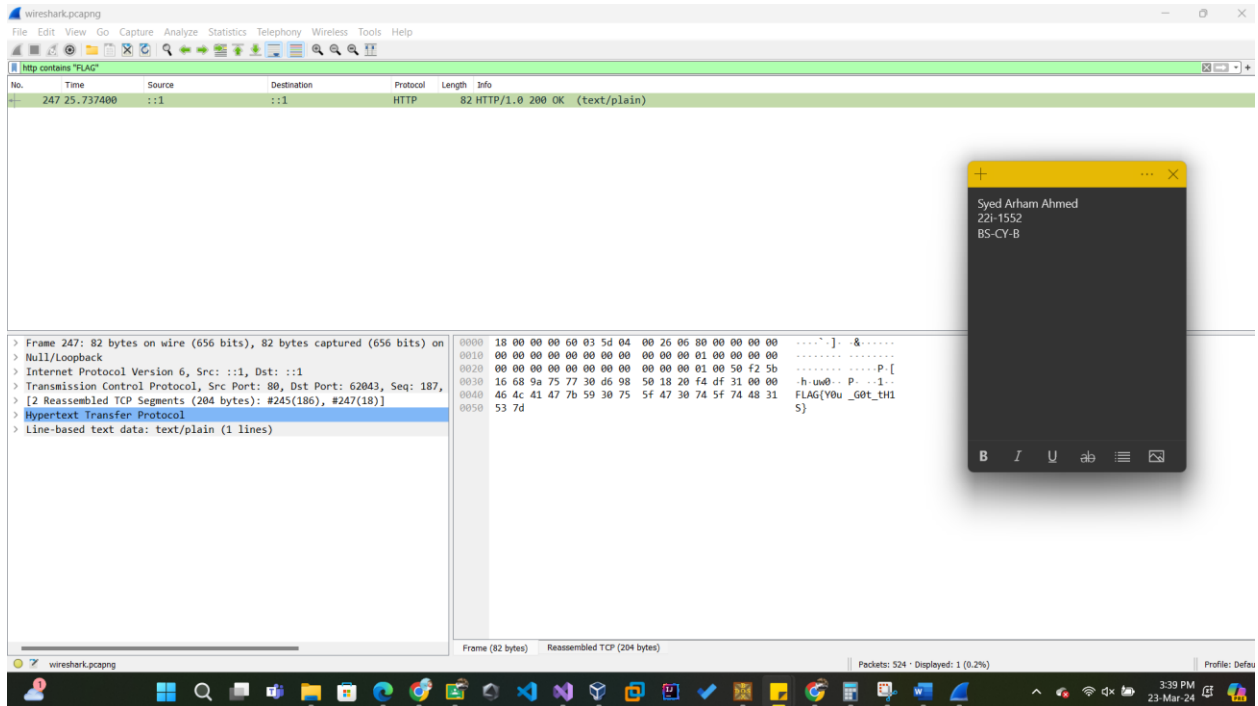
Well, the static image URL did not generate many packets but the whole website generated thousands of packets as a lot more data needed to be loaded other than just still images with no embeds.



## Q.2.

http contains "FLAG" {This command searches for the word FLAG in the TCP packets}

I ran the command listed above and it gave me the packet ie:247 which contains the flag.



Q.3.

The screenshot displays a web browser window showing a LinkedIn profile for Arham Ahmed. The profile includes a profile picture, a cover image, and text identifying him as a student at the National University of Computer and Emerging Sciences. The browser's developer tools are open on the right, showing a network request to a LinkedIn endpoint with a 200 OK status. The Windows taskbar at the bottom shows the system clock as 11:32 PM on 16-Mar-24.

**LinkedIn Profile Details:**

- Name:** Arham Ahmed
- Education:** Student of Cyber Security at National University of Computer and Emerging Sciences, Islamabad, Pakistan
- Followers:** 202
- Connections:** 200
- Profile Language:** English
- Public profile & URL:** www.linkedin.com/in/arham-ahmed-090a64270

**Developer Tools Network Log:**

Name	Request URL	Request Method	Status Code	Remote Address	Referrer Policy
A17077741113937e=17112...	https://www.linkedin.com/sensorCollect/?action=reportMetrics	POST	200 OK	13.107.42.14443	strict-origin-when-cross-origin

The screenshot shows a web browser window with the address bar displaying `linkedin.com/404/?_l=en_US`. The main content area shows a "Page not found" message from LinkedIn, stating: "Uh oh, we can't seem to find the page you're looking for. Try going back to the previous page or see our [Help Center](#) for more information." Below this message is a button labeled "Go to your feed".

The browser's developer tools are open on the right side, showing the Network tab. The filter is set to "Fetch/XHR". The list of requests shows a 404 error. The console shows a message: "Log XMLHttpRequests" and "Eager evaluation".

The screenshot displays a web browser with two tabs: 'Arham Ahmed | LinkedIn' and 'Feed | LinkedIn'. The Chrome DevTools interface is open, showing the 'Network' panel with a list of requests. The selected request is 'voyagerSocialDashReaction...', which is a POST request to 'https://www.linkedin.com/voyager/api/voyagerLegoDashWidgetImpressionEvents' with a status code of 201 Created. The 'Headers' tab is active, showing request and response details. Below the Network panel, the 'Console' panel is visible, showing a summary of 1,290 messages, 3 user messages, 464 errors, 60 warnings, 756 info, and 10 verbose logs. The Windows taskbar at the bottom shows the time as 11:36 PM on 16-Mar-24.

Arham Ahmed | LinkedIn x Feed | LinkedIn x + - [Icons] [All Bookmarks]

Elements Console **Network** >> 464 60 56 [Settings] [Close]

[Icons] [Filter] [Invert] [Hide data URLs] [Hide extension URLs]

All **Fetch/XHR** Doc CSS JS Font Img Media Manifest WS Wasm Other

☐ Blocked response cookies ☐ Blocked requests ☐ 3rd-party requests

20000 ms 40000 ms 60000 ms 80000 ms 100000 ms 120000 ms 140000 ms

Name X **Headers** Payload Preview Response >>

▼ General

Request URL: https://www.linkedin.com/voyager/api/voyagerLegoDashWidgetImpressionEvents

Request Method: POST

Status Code: 201 Created

Remote Address: 13.107.42.14:443

Referrer Policy: strict-origin-when-cross-origin

▼ Response Headers

816 / 1124 requests | 1.0 MB / 2.3

Console Issues [Icons] [Filter] Default levels 3 Issues: 56 2 [Settings]

1,290 messages

3 user messages

464 errors

60 warnings

756 info

10 verbose

☐ Hide network ☒ Log XMLHttpRequests

☐ Preserve log ☒ Eager evaluation

☐ Selected context only ☒ Autocomplete from history

☒ Group similar messages in console ☒ Treat code evaluation as user action

☒ Show CORS errors in console

11:36 PM 16-Mar-24 [Icons]

The screenshot displays a web browser window with a YouTube video titled "4x4 Dangal" by Abdul VahabVaince. The video has 10K views and was posted 1 day ago. The video content shows a black Mitsubishi Pajero driving on a dirt road with other vehicles in the background. The browser's address bar shows the URL "the cybertraveller22 (Syed Arham Ahmed) (561) YouTube".

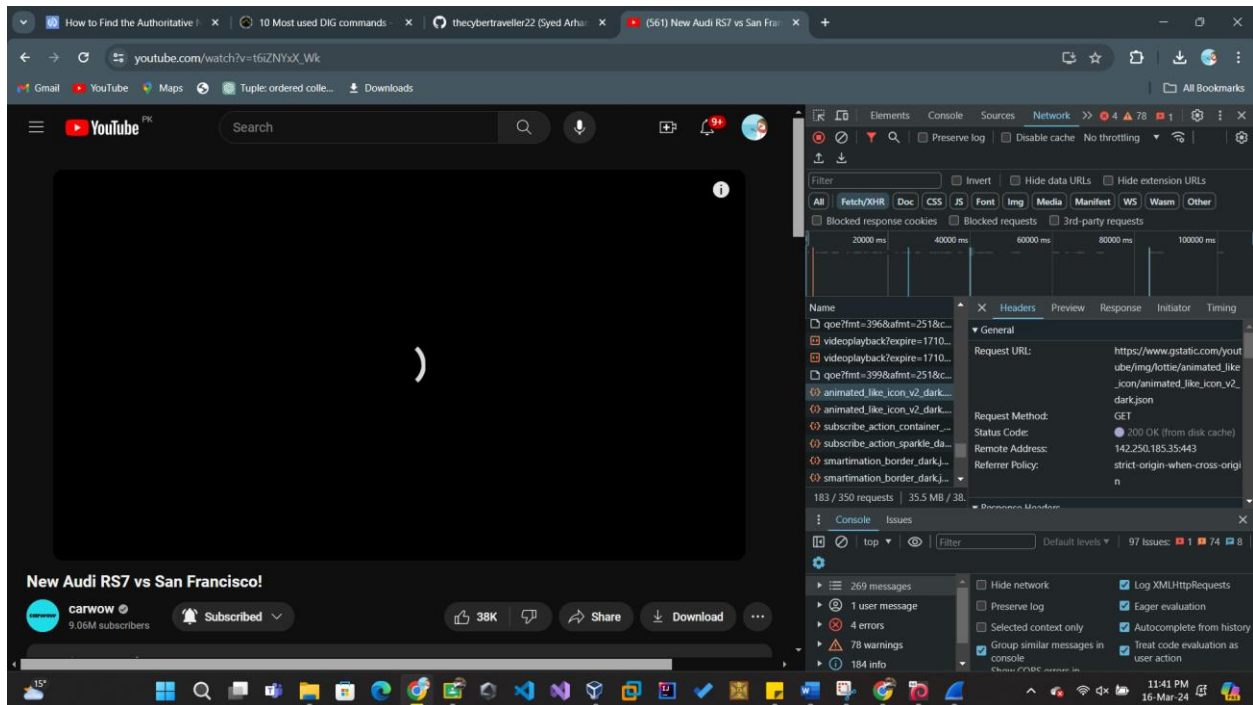
The Chrome DevTools Network panel is open, showing a list of network requests. The selected request is "log\_event?alt=json&key=AL...", which is a POST request to "0.177:0.000&bat=0.177:1:18&v=0.177:5&bh=0.177:0.000". The response status is "204 No Content". The response headers show "Access-Control-Allow-Credentials: true".

The Console panel is also open, showing 97 issues. The messages list includes "desktop\_poly...", "www.youtub...", "base.js", and "web-animations...".

The bottom of the screen shows the Windows taskbar with various application icons and the system clock displaying "11:37 PM 16-Mar-24".



## Syed Arham Ahmed\_22i-1552\_BS-CY-B\_CNET\_Assignment#2



The above image shows us that it got the data from the cache instead of asking the DNS server.

